ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO

Tarefa 2 - MAP3121

Fórmulas de Integração Numérica de Gauss

Arthur de Azevedo e Almeida Maia — 11819921
Caio Nascimento Balreira — 11805342

São Paulo, SP
2022

1. Introdução

Nesta tarefa, foi solicitada a implementação de um programa para o cálculo de integrais duplas por meio das Fórmulas de Integração de Gauss.

No estudo dessas fórmulas, encontra-se a seguinte fórmula para aproximar a integração de uma função f por meio de uma integração numérica:

$$\int_a^b f(x)\,dx = \sum_{j=1}^{n} \omega_j f(x_j) + E_n(f) \tag{1}$$

onde os nós xj ∈ [a, b] e os pesos ωj são determinados impondo-se condições para o erro E(f). No caso do método de Gauss, E(f) = 0 e os pesos e nós têm valores já conhecidos para o intervalo de integração de [-1,1]. Para um intervalo [a,b] qualquer, é necessário realizar uma transformação de variáveis para que trabalhemos novamente com o intervalo [-1,1]. A transformação é como mostrada abaixo:

$$x = A \cdot \xi + B$$

Onde ξ é a nova variável para a qual o intervalo da integral passa a ser [-1,1]. Sabendo disso e do fato de x pertencer a [a,b], conseguimos obter os valores para A e B:

$$A = \frac{b-a}{2}$$

$$B = \frac{b+a}{2}$$

Com isso, a fórmula da integração de Gauss fica:

$$\int_a^b f(x)\,dx \approx \frac{b-a}{2}\sum_{i=1}^{n} w_i f\left(\frac{b-a}{2}\xi_i + \frac{a+b}{2}\right) \tag{2}$$

Onde os pesos e nós que são usados são os já conhecidos para o valor de n escolhido.

Há uma limitação, no entanto, no grau de f(x). O grau de de f(x) deve ser inferior a 2n. Pode-se observar isso ao notar que caso f(x) fosse um polinômio de grau 2n e a equação (1) fosse exata (isto é, E(x) = 0), o polinômio dado por:

$f(x) = (x - x_1)^2 (x - x_2)^2 ... (x - x_n)^2$ teria resultado igual a zero para o somatório pois x1 , x2 … xn seriam as raízes do somatório; o que é contraditório, pois sua integral é positiva. Assim, caso exista uma fórmula como (1) que seja exata, para polinômios de grau menor ou igual a 2n − 1, temos:

$$\int_a^b (x - x_1)(x - x_2)\ldots(x - x_n)q(x)\,dx = 0$$

para todo q(x) com grau inferior a n, uma vez que o somatório teria x1, x2,...,xn como raízes.

Levando em conta os desenvolvimentos acima, as integrais duplas podem ser resolvidas numericamente como mostrado abaixo:

$$I = \int_a^b \left[\int_{c(x)}^{d(x)} f(x, y)\, dy \right] dx$$

$$I = \sum_{i=1}^{n} u_i F(x_i) \qquad \text{com} \qquad F(x_i) = \sum_{j=1}^{n} v_{ij} f(x_i, y_{ij}),$$

onde xi e ui são os nós e os pesos no intervalo [a, b], e yij e vij são os nós e os pesos nos intervalos [c(xi), d(xi)].

2. Objetivo

O objetivo deste exercício é a elaboração de um programa computacional que seja capaz de solucionar integrais duplas, em um intervalo sem extremos no infinito, de uma função qualquer utilizando o método numérico de integração de Gauss.

Para testar o funcionamento do código, devem ser resolvidos, utilizando o programa desenvolvido, 4 exemplos disponibilizados no enunciado da tarefa.

3. Desenvolvimento

O código desenvolvido foi baseado no pseudocódigo apresentado no livro *Análise Numérica - Richard L. Burden e J. Douglas Faires*. A função apresentada no livro, quando “traduzida” para Python, resultou no seguinte código:

```python
def dupla(a, b, n, c, d, f):
    h1 = (b-a)/2
    h2 = (b+a)/2
    J = 0

    for i in range(len(n)):
        JX = 0
        r1 = n[i][0]
        w1 = n[i][1]
        x = h1*r1 + h2
        c1 = c(x)  # definir c(x) para cada exemplo
        d1 = d(x)  # definir d(x) para cada exemplo
        k1 = (d1 - c1)/2
        k2 = (d1 + c1)/2

        for j in range(len(n)):
            r2 = n[j][0]
            w2 = n[j][1]
            y = k1*r2 + k2
            Q = f(x, y)
            JX = JX + w2*Q

        J = J + w1*k1*JX

    J = h1*J
    return(J)
```

onde f é a função f(x, y) a ser integrada, d é o limite de integração superior da integral interna, c é o limite inferior desta, b é o limite de integração superior da integral externa, a é o limite inferior desta e n é o conjunto de pontos e pesos utilizados no método, da seguinte maneira:

$$\int_a^b \int_{c(x)}^{d(x)} f(x, y)\, dy\, dx$$

Para o cálculo de cada um dos casos pedidos no enunciados, evitou-se declarar novas funções e, em vez disso, optou-se por passar como parâmetros da função, funções do tipo *lambda*. Este tipo de função presente no Python é um tipo de função anônima, que serve para não precisar realizar a declaração de funções com a palavra reservada “def” no início do código. Assim, por exemplo, a chamada da função da integral dupla que calcula o volume do cubo de lado 1 (como solicitado no exemplo 1) para n = 6 ficou:

```
[dupla(0, 1, n6, lambda x: 0, lambda x: 1, lambda x, y: 1)
```

A única exceção para a utilização de uma função lambda no lugar de uma função declarada foi para a área pedida no terceiro exemplo. Como a expressão para esta era grande, optou-se por declarar esta função e passá-la como parâmetro para a função que calcula a integral dupla, da seguinte forma:

```
# Função utilizada para uma das integrais:
def zExpArea(x, y):
    return ((e**(y/x) / x)**2 + (y*e**(y/x) / x**2)**2 + 1)**0.5
```

Com que a chamada da função fica, por exemplo:

```
dupla(0.1, 0.5, n10, lambda x: x**3, lambda x: x**2, zExpArea)]
```

4. Funcionamento

Como citado na sessão anterior, o programa resolve cada um dos quatro exemplos pedidos no enunciado para n = 6, n = 8 e n = 10. Segue abaixo o menu apresentado ao executar o programa:

```
Bem-vindo ao programa de integração numérica de Gauss escrito por
Arthur Maia e Caio Balreira.

Neste programa, existem 5 funcionalidades:
 1) Calcular o volume do cubo cujas arestas tem comprimento 1 e o
do tetraedro com vértices em (0,0,0), (1,0,0), (0,1,0), (0,0,1)
 2) Calcular a área da região no primeiro quadrante limitada pelos
 eixos e pela curva y = 1 - x^2
 3) Calcular a área e volume abaixo da superfície descrita por z =
 e^(x/y), 0.1 <= x <= 0.5, x^3 <= y <= x^2
 4) Calcular o volume da calota esférica de altura 1/4 da esfera d
e raio 1 e o do sólido obtido da rotação da região, em torno do ei
xo y, delimitada por x = 0, x = e^(-y^2), y = -1 e y = 1
 0) Encerrar o programa

Escolha a funcionalidade que deseja executar: 
```

Exemplo 1: O volume do cubo cujas arestas têm comprimento 1 é dado por:

$$V_{cubo} = \int_0^1 \int_0^1 1 \, dy \, dx$$

Cujo resultado é igual a 1 u. v.

Já para o tetraedro, o volume é dado por:

$$V_{tetraedro} = \int_0^1 \int_0^{-x+1} 1 - x - y \, dy \, dx$$

Cujo resultado vale ⅙ u.v. (ou 0,167 aproximadamente).

Ao executar a funcionalidade 1 do programa, a resposta é a seguinte:

```
Escolha a funcionalidade que deseja executar: 1

O volume do cubo é:
    Para n6: 1.0 u.v.
    Para n8: 1.0 u.v.
    Para n10: 0.9999999999999998 u.v.
    Valor esperado: 1 u.v.

O volume do tetraedro é:
    Para n6: 0.16666666666666663 u.v.
    Para n8: 0.16666666666666669 u.v.
    Para n10: 0.16666666666666669 u.v.
    Valor esperado: 0.16666666... u.v.
```

Cujos resultados são exatos, a não ser por problemas de aproximação. Como a integração de Gauss leva em conta um erro E(x) = 0, é esperado, como mostrado acima, que os resultados da integração numérica sejam exatos.

Exemplo 2: Neste exemplo, a integral dupla a ser resolvida é dada por:

$$A = \int_0^1 \left[\int_0^{1-x^2} dy \right] dx = \int_0^1 \left[\int_0^{\sqrt{1-y}} dx \right] dy = \frac{2}{3}$$

Que corresponde a área da região no primeiro quadrante limitada pelas curvas dadas no enunciado.

Ao executar a funcionalidade 2 do programa, obtém-se:

```
Escolha a funcionalidade que deseja executar: 2

A área em baixo da parábola, para o limite de 1 - x^2, no interval
o pedido é:
    Para n6: 0.6666666666666667 u.a.
    Para n8: 0.6666666666666666 u.a.
    Para n10: 0.6666666666666665 u.a.
    Valor esperado: 0.6666666... u.a.

A área em baixo da parábola, para o limite de √(1 - y), no interva
lo pedido é:
    Para n6: 0.6670464379156135 u.a.
    Para n8: 0.6668355801001765 u.a.
    Para n10: 0.6667560429365086 u.a..
    Valor esperado: 0.6666666... u.a.
```

Que, como mostrado na imagem, condizem com o resultado esperado.

<u>Exemplo 3</u>: Neste caso, para o cálculo da área solicitada, precisamos utilizar a expressão:

$$\iint_R \sqrt{f_x(x,y)^2 + f_y(x,y)^2 + 1}\, dx\, dy$$

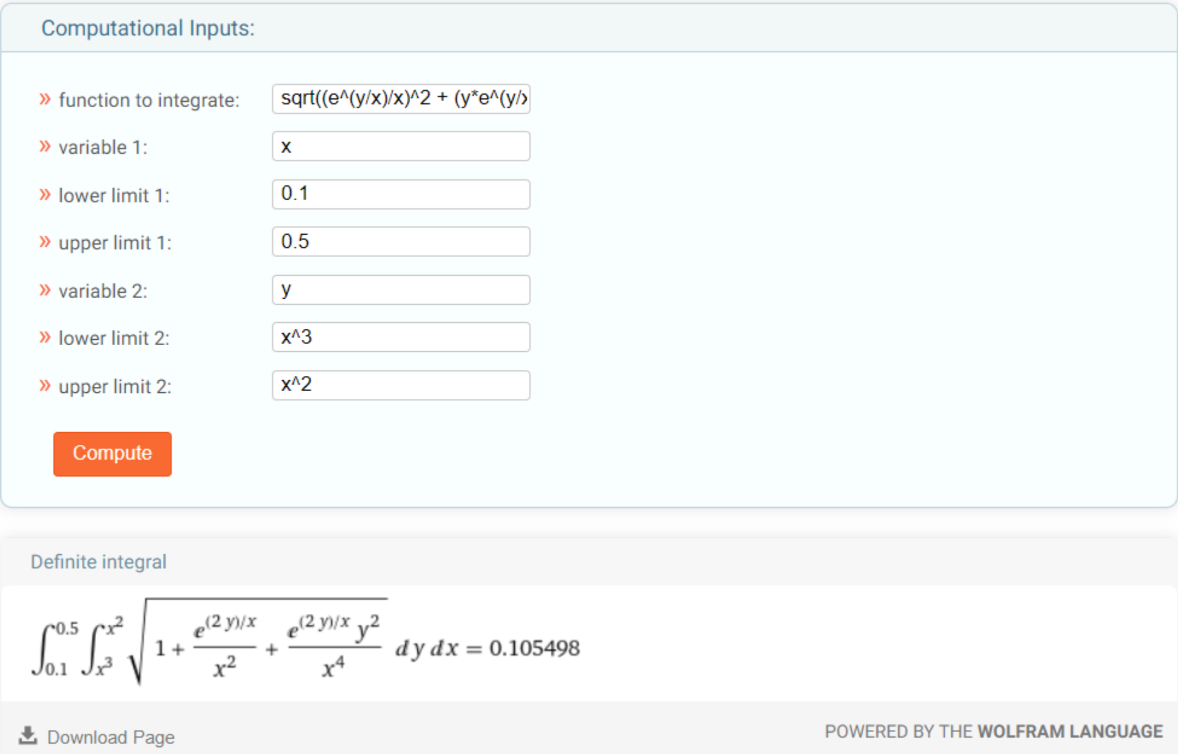
Onde $f_x(x, y)$ e $f_y(x, y)$ são respectivamente as derivadas parciais em função de x e de y da função f(x, y), que neste caso é $f(x, y) = e^{y/x}$.

Logo, $f_x(x, y) = \frac{-y}{x^2}e^{y/x}$ e $f_y(x, y) = \frac{1}{x}e^{y/x}$.

Assim, a área é dada por:

$$\int_{0.1}^{0.5}\int_{x^3}^{x^2} \sqrt{f_x(x, y)^2 + f_y(x, y)^2 + 1}\, dy\, dx$$

Cujo resultado foi obtido com auxílio da ferramenta Wolfram Alpha como mostrado abaixo:



Já o volume é calculado por:

$$V = \int_{0.1}^{0.5}\int_{x^3}^{x^2} e^{y/x} dy\, dx$$

Cujo resultado é aproximadamente 0,0333056 u.v..

Ao executar a funcionalidade 3 do programa, obtém-se:

```
Escolha a funcionalidade que deseja executar: 3

O volume em baixo da superfície no intervalo pedido é:
    Para n6: 0.03330556611623719 u.v.
    Para n8: 0.03330556611623208 u.v.
    Para n10: 0.033305566116232074 u.v.
    Valor esperado: 0.0333055661162 u.v.

A área dá superfície no intervalo pedido é:
    Para n6: 0.10549788240049787 u.a.
    Para n8: 0.10549788240051994 u.a.
    Para n10: 0.10549788240051994 u.a.
    Valor esperado: 0.105497882401 u.a.
```

Cujo resultado é de acordo com o esperado.

<u>Exemplo 4</u>: Neste caso, utiliza-se a seguinte integral dupla para o cálculo do volume do sólido de revolução:

$$V = 2\pi \iint_R d_\gamma(x, y)\, dx\, dy$$

Onde d_γ é a distância do ponto até o eixo em volta do qual a área está sendo rodada. Ou seja, é a própria coordenada x do ponto (a distância de um ponto ao eixo y é a sua coordenada x). Contudo, como a função acima calcula a integral dupla na ordem dx dy enquanto a função desenvolvida a calcula na ordem dy dx, as variáveis foram trocadas de tal forma que o volume de revolução em si não foi alterado, mas apenas a orientação do gráfico. Por exemplo, o gráfico de $x = e^{-y^2}$ passou a ser o de $y = e^{-x^2}$, conforme os gráficos abaixo, enquanto o eixo de rotação passou a ser x, e não y e, portanto, a distância do gráfico até o eixo de rotação é seu valor de y, não mais o de x.

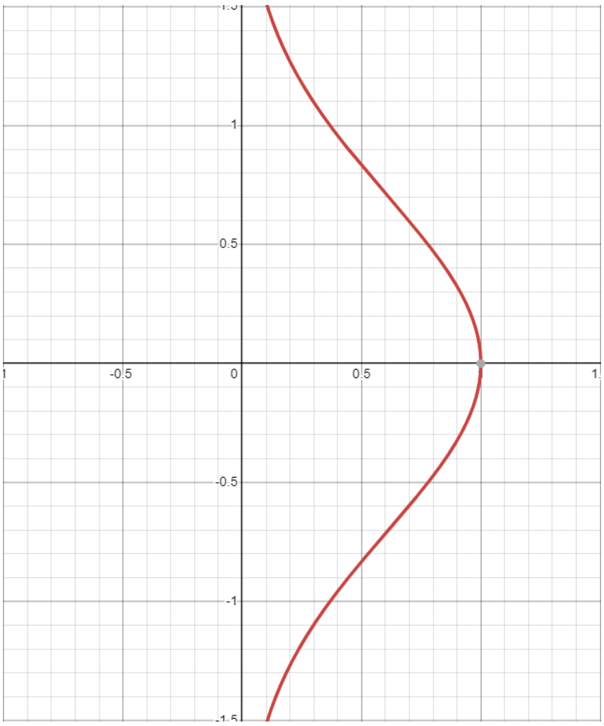


Gráfico de $x = e^{-y^2}$

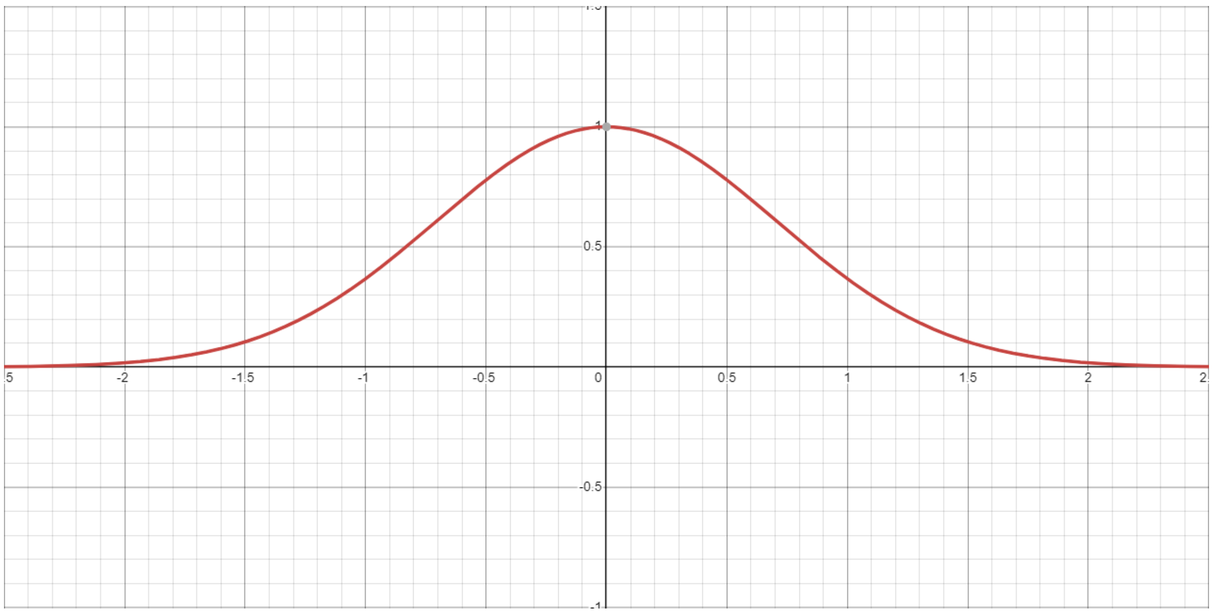


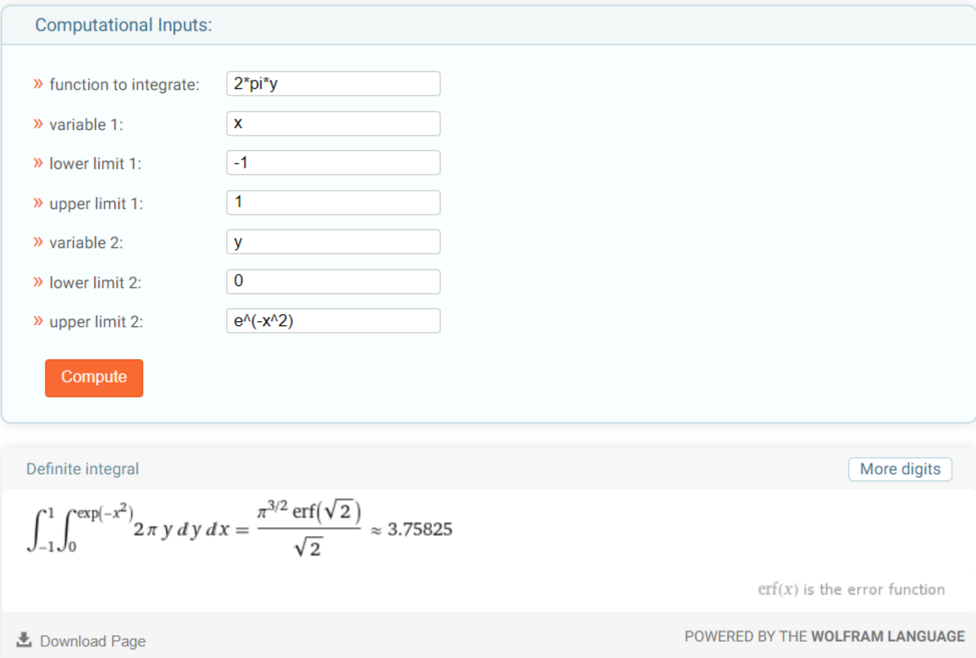
Gráfico de $y = e^{-x^2}$

Os valor esperado para o volume da calota esférica foi obtida conforme a fórmula

$$\frac{1}{3}\pi h^2(3r - h),$$

onde h é a altura da calota (neste caso, 0,25 u.c.) e r é o raio da esfera, neste caso 1 u.c. Substituindo os valores, tem-se que V ≈ 0,179987079112 u.v.

Enquanto, para o volume do outro sólido de revolução, o valor esperado foi obtido calculando a integral de revolução com auxílio da ferramenta Wolfram Alpha como mostrado abaixo:



Ao executar a funcionalidade 4 do programa, obtém-se:

```
Escolha a funcionalidade que deseja executar: 4

O volume da calota esférica é:
    Para n6: 0.1799870791119152 u.v.
    Para n8: 0.17998707911191522 u.v.
    Para n10: 0.17998707911191525 u.v.
    Valor esperado: 0.179987079112 u.v.

O volume do sólido de revolução é:
    Para n6: 3.7581650328967093 u.v.
    Para n8: 3.7582492624394384 u.v.
    Para n10: 3.7582496332093873 u.v.
    Valor esperado: 3.75824963423 u.v.
```

Cujo resultado é de acordo com o esperado.

Referências

Enunciado do EP2 <tarefa2_2022.pdf (usp.br)>. Acessado em 05/06/2022

BURDEN, Richard; FAIRES, Douglas; BURDEN, Annette. **ANÁLISE NUMÉRICA**. 10ª edição.

Gaussian Integration(Derivation of 2 point and 3 point formula) by Keshav Jadhav <(111) Gaussian Integration(Derivation of 2 point and 3 point formula) by Keshav Jadhav - YouTube>. Acessado em 05/06/2022

Quadratura Gaussiana + Algoritmo em Python <(111) Quadratura Gaussiana + Algoritmo em Python - YouTube>. Acessado em 05/06/2022

Calota Esférica <Calota esférica – Wikipédia, a enciclopédia livre (wikipedia.org)>. Acessado em 05/06/2022

Wolfram Alpha <Wolfram|Alpha: Computational Intelligence (wolframalpha.com)>

Anexos

Repositório onde os códigos e os arquivos LEIAME são armazenados: <https://github.com/ArthurAAM/EPNumerico>